



# A Smarter Solution for Better Quality

*The New IBM Rational Quality Management Solution*

Sreenivasan Rajagopal  
Product Manager, IBM Rational Test Virtualization Solution  
rajagops@us.ibm.com

IBM Software

# Innovate2012

The Premier Event for Software and Systems Innovation

Next  NOW!

## Please Note:

IBM's statements regarding its plans, directions, and intent are subject to change or withdrawal without notice at IBM's sole discretion.

Information regarding potential future products is intended to outline our general product direction and it should not be relied on in making a purchasing decision.

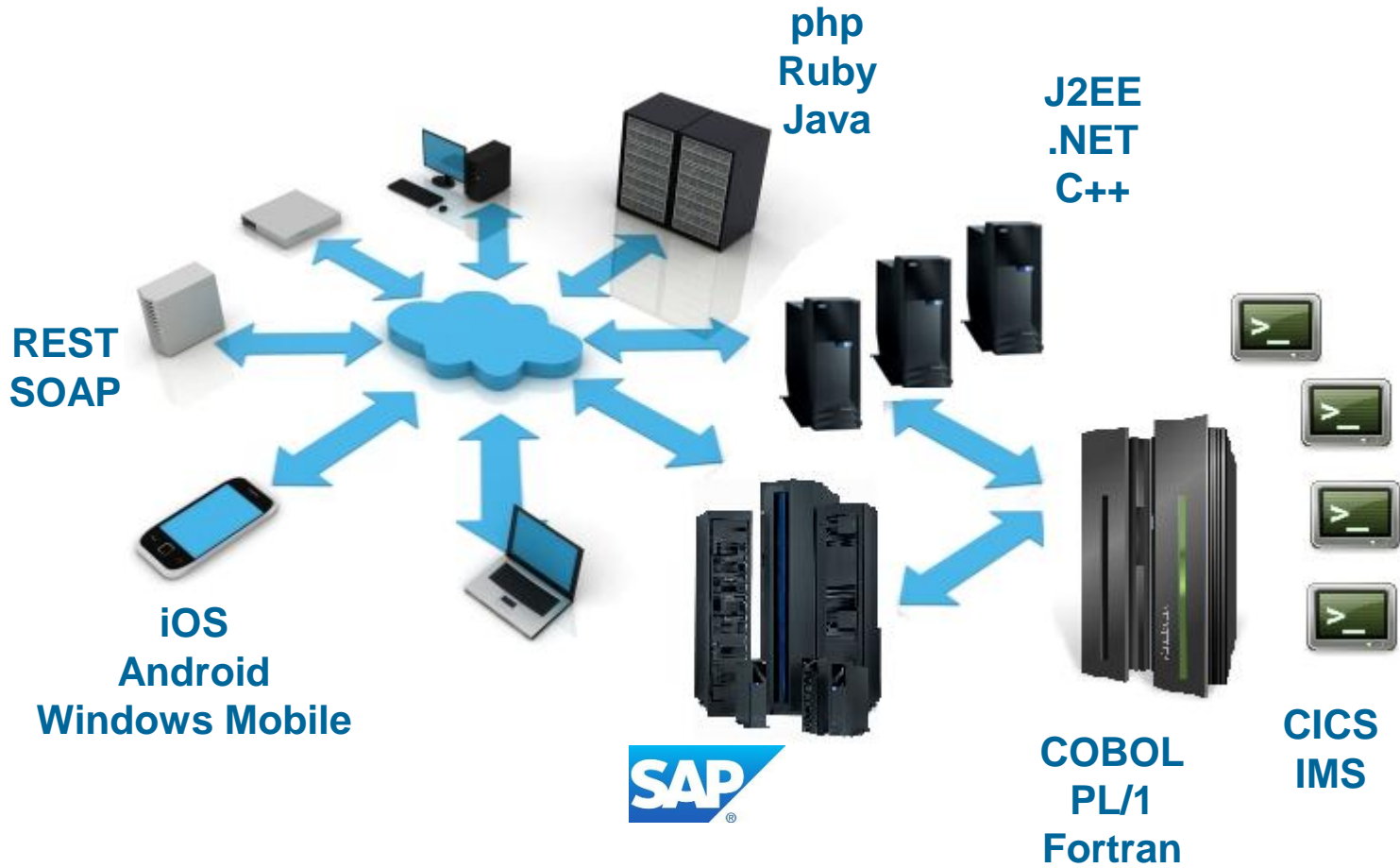
The information mentioned regarding potential future products is not a commitment, promise, or legal obligation to deliver any material, code or functionality. Information about potential future products may not be incorporated into any contract. The development, release, and timing of any future features or functionality described for our products remains at our sole discretion.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput or performance that any user will experience will vary depending upon many factors, including considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve results similar to those stated here.

# 40 years of technology...



# Powers today's cutting edge IT solutions



# Software drives today's innovation for a smarter planet

*Transforming the way we live, work, and play*





## Yet software quality is a major problem across all industries

- Software is blamed for **more major business problems than any other man-made product.**
- Poor software quality has become **one of the most expensive topics in human history**
  - **\$150+ billion per year in U.S.**
  - **\$500+ billion per year worldwide.**
- Projects cancelled due to poor quality are **15% more costly than successful projects** of the same size and type.



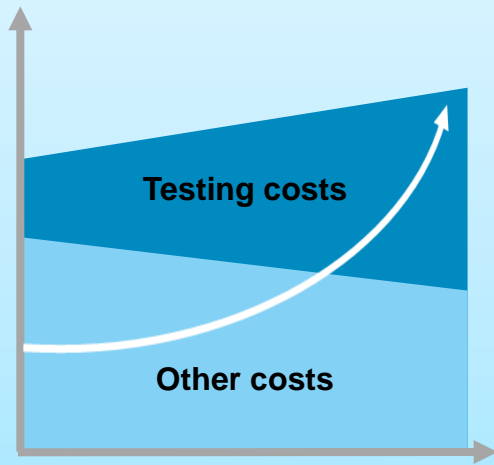
Source: Capers Jones, 2011

Based on 675 companies, 35 government/military groups, 13,500 projects, 50-75 new projects/month, 24 countries, 15 lawsuits

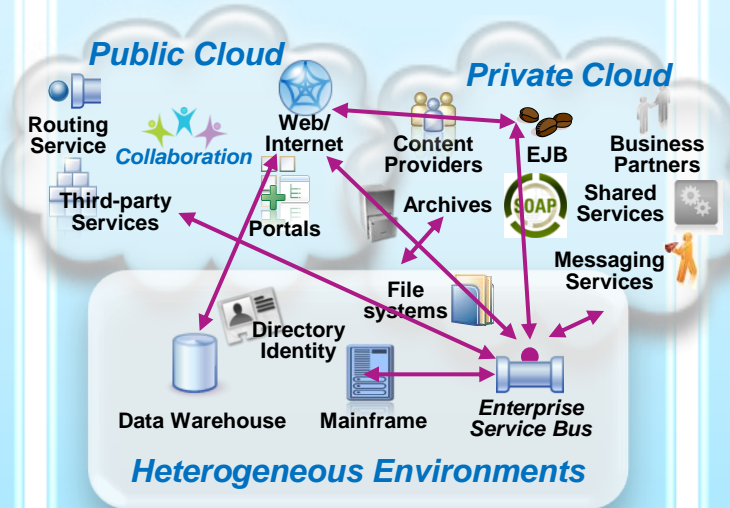
# Cost, complexity and velocity make today's quality paradigm impractical

*An estimated 60 - 80 percent of the cost of software development is in rework\**

## Increasing Cost of Quality



## Increasing Development Complexity



## Balancing Quality and Speed



Outsourcing **labor** is no longer a sustainable model as global wages are increasing

Product and application **complexity** and size are increasing

**Productivity is inhibited** as test teams can no longer keep up with agile development

\* Source: <http://www.sei.cmu.edu/about/message/>

# Managing software quality has become extremely challenging

## Increasing Cost of Quality

**\$59.5 billion**

The estimated cost of software defects cost to U.S. economy<sup>c</sup>

**13%**

The forecasted increase in wages for India IT workforce in 2011<sup>a</sup>

## Increasing Development Complexity

**\$5-30 million**

The typical investment to build a single test lab for a Fortune 500 company. Most have

**dozens<sup>b</sup>...  
5X more test labs required**

Large global financial services firm increases test lab deployments from 6 to 32 in 13 years<sup>b</sup>

## Balancing Quality and Speed

**74%**

The estimated number of projects with significant delays or quality issues<sup>c</sup>

**30-50%**

The average amount of time testing teams spend on setting up test environments, instead of testing<sup>c</sup>

<sup>a</sup> The Times of India, IT sector to get 12% average salary hike in 2011, TOI Tech & Agencies, March 8, 2011.

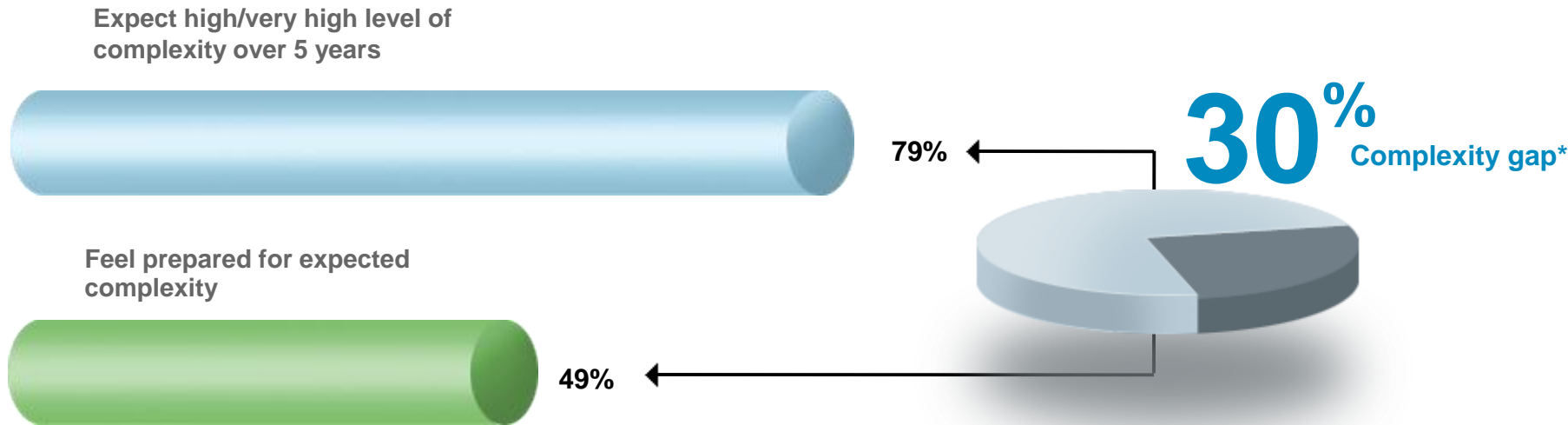
<sup>b</sup> IBM customer reference.

<sup>c</sup> NIST, Planning Report 2002-2003. The Economic Impacts of Inadequate Infrastructure for Software Testing, May 2002.



# Complexity is cited as primary challenge in 2010 Global CEO Study

## Expected level of complexity and preparedness to handle



***“Complexity should not be viewed as a burden to be avoided; we see it as a catalyst and an accelerator to create innovation and new ways of delivering value.”***

Juan Ramon Alaix  
President, Pfizer Animal Health, United States

\* Complexity gap = difference between expected complexity and the extent to which CEOs feel prepared to manage complexity

Source: Q10 How much complexity will your organization have to master over the next 5 years? n=1,512; respondents who selected “More/significantly more complexity”; Q12 How prepared do you feel for the expected complexity ahead? n=1,508; respondents who selected “Fully prepared”

# Many businesses are challenged by growing complexity

**8 in 10** CEOs anticipate significant complexity ahead.



79%

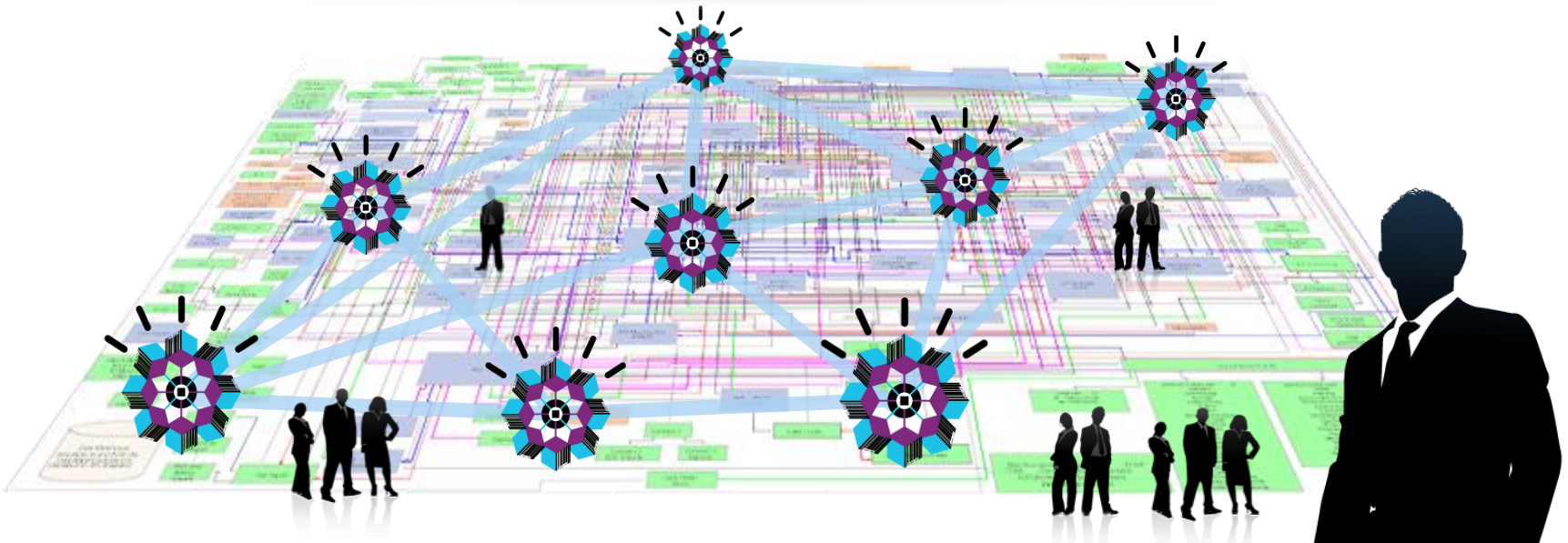
**Half** Feel prepared to handle it.



49%

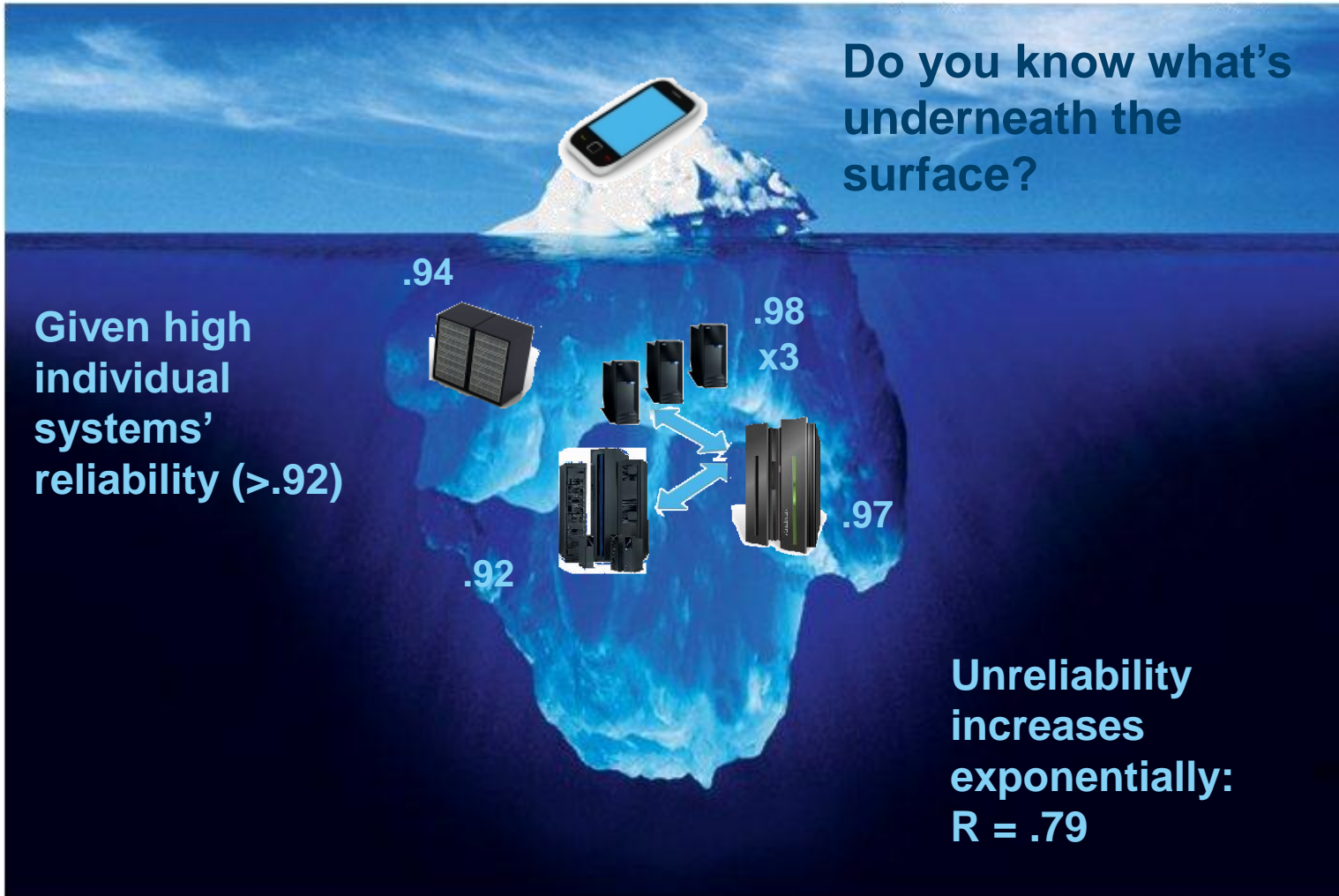
**30%**  
complexity gap

Source: IBM Global CEO Study 2010  
"Capitalizing on complexity".



**Growing Complexity**

# And yet, we test the same way



# It is not about the application... It's about the process!



Bid on your fantasy vintage computer

Check email



You've been outbid

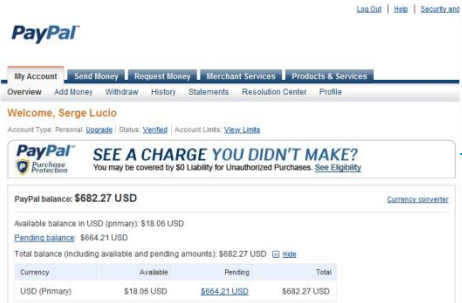
Bid again

You get Outbid

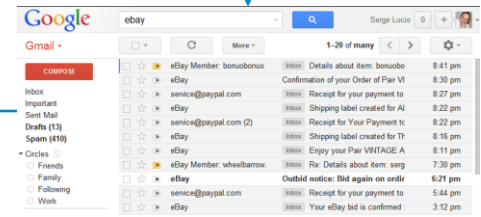


Track shipment on your iPhone

Seller ships



Check out on your PC



You won!

## The user has changed

Today's consumers ...

- Have a deeper knowledge and acceptance of technology
- Want things faster, cheaper, better, etc...
- Are willing to try the “latest” & “greatest” gadget
- Open to risk but the product must deliver value
- Determine who will win the marketplace
- Demand a higher level of quality
  - but accepting of defects provided issues are quickly resolved





## So, what if you could...

- start testing long before the UI is available?
- stand up a test environment in minutes?
- test every build for regressions?
- test any system, irrespective of technology, using the same solution?
- collaborate more effectively with developers?
- do more exploratory testing and automate regression testing?

# What if you could shift from testing to quality?

## Testing

A technical investigation done to expose quality-related information about the product or service under test



*"I have hundreds of testers & lots of automation, but all I do is find more defects. I don't have a testing problem, I have a quality problem." – large global bank*

## Quality Management

Systematic monitoring and evaluation of the various aspects of a product or service, to maximize the probability that target quality standards are being attained

### Chaotic

- Catch & patch
- Ad-hoc testing

0

### Repeatable

- Test planning
- Test automation

1

### Proactive

- Collaborative quality assurance
- Broader scope
  - Security
  - Compliance
  - Accessibility

2

### Prevention

- Integration test
- Testability, maintainability
- Defined policies and quality gates
- Automated regression

3

### Optimization

- Ent. Quality Management
  - Gap analysis
  - Risk assessments
- Process improvements

4

# The New IBM Rational Solution

## *Powered by Green Hat Technology*



### ■ IBM Rational Test Workbench

- Enable functional, performance and integration testing throughout your project lifecycle.
- With a script less, wizard-driven test authoring environment and support for more than 70 technologies and protocols,

### ■ IBM Rational Performance Test Server

- Save time and effort by reusing your functional integration tests for performance testing
- Combine your tests to model real-world scenarios and assess the impact of load on your integrated infrastructure.

### ■ IBM Rational Test Virtualization Server

- Model real system behavior to eliminate test dependencies and simulate production,
- Accelerate testing and reduce the setup and infrastructure costs of traditional testing environments.



QA staff

Test Workbench	Functional Testing Integration Testing Performance Testing Regression Testing
----------------	--

Test Servers	Load Agents Virtualization Agents
--------------	--------------------------------------

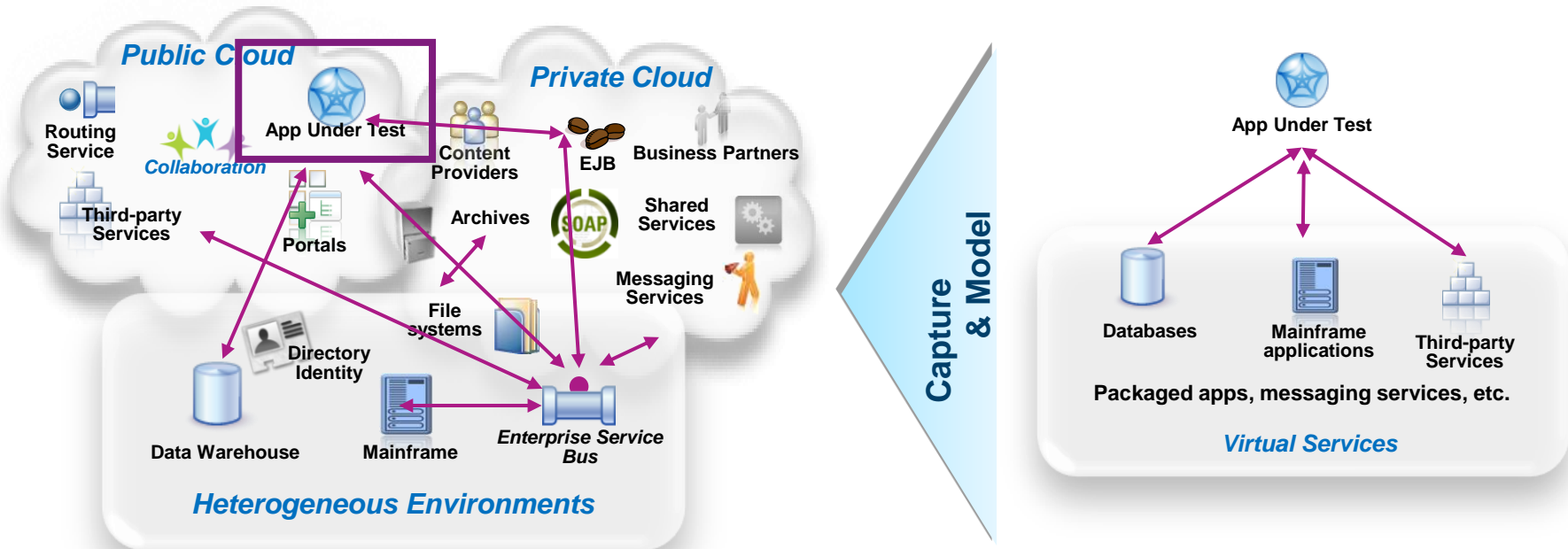


Test Lab Infrastructure

*Launched June 2012*

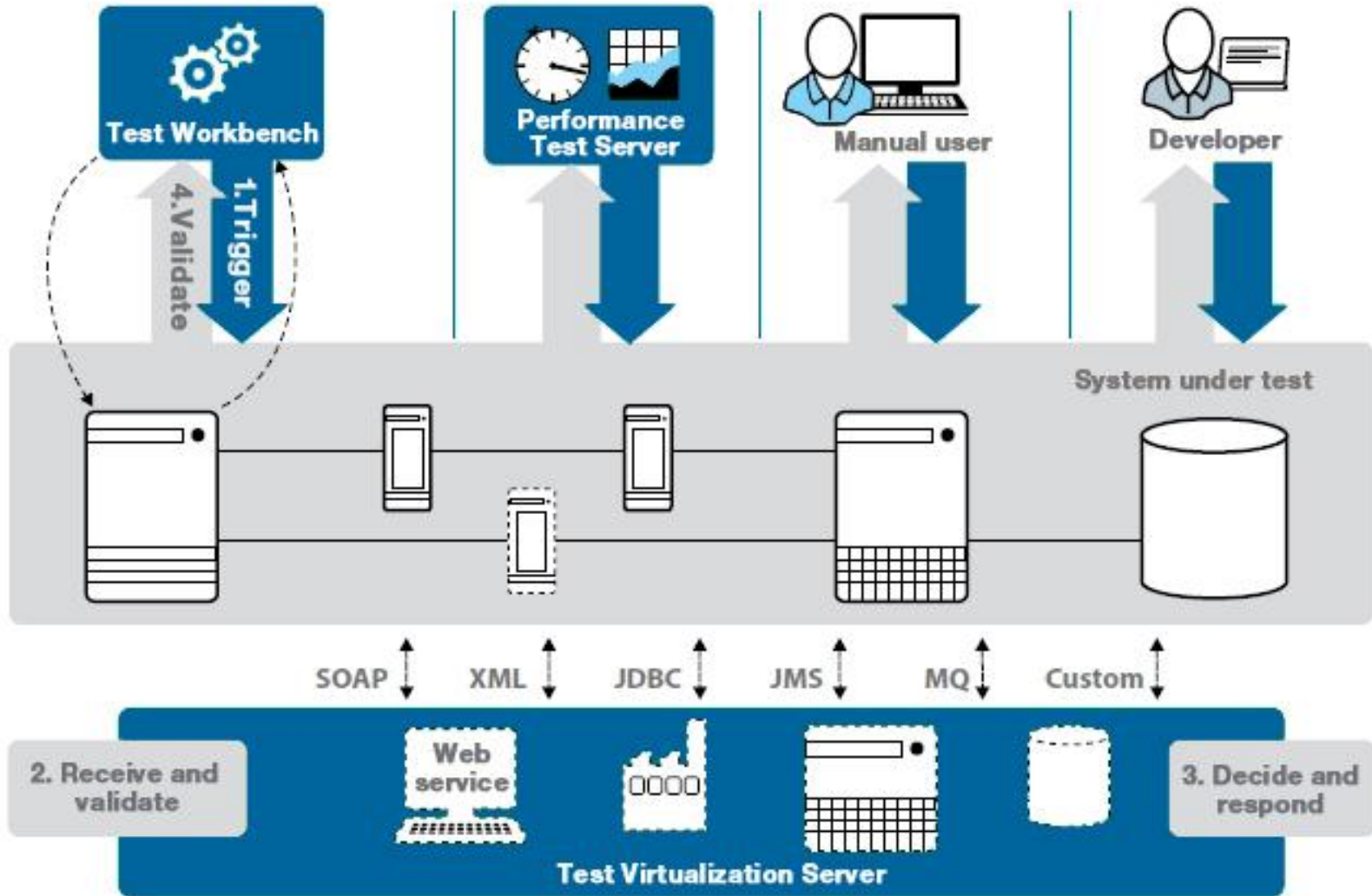


# Isolate your subsystems with Test Virtualization



- Test Virtualization enables to simulate the behavior of an entire application or system during testing
- Virtual Services can run on commodity hardware, private cloud, public cloud to provide flexible test environments
- Developer and testers continue to use their testing tools

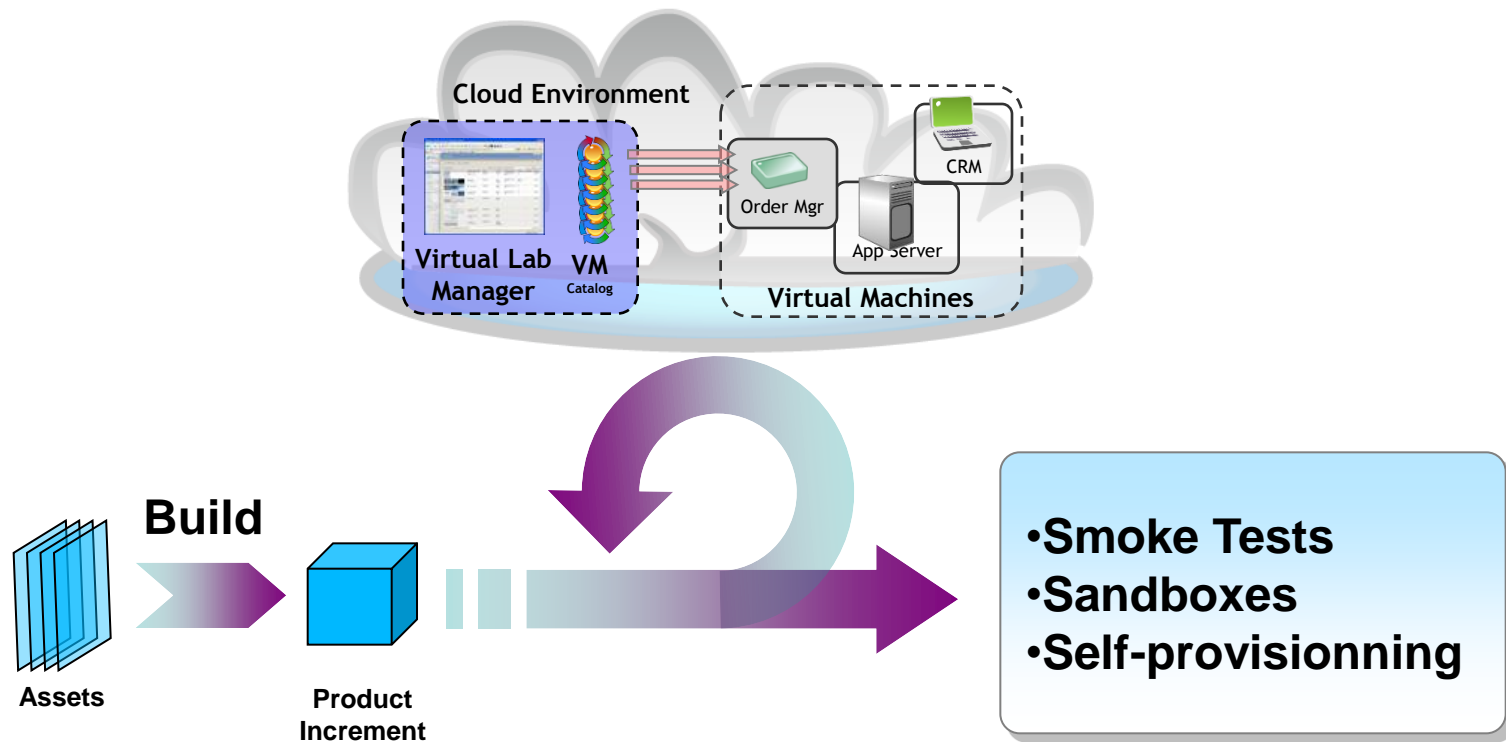
# Automate your tests at any layer of your application





# Streamline your process with continuous integration & test

- Avoid time spent installing and configuring software just to discover basic build issues
- Smoke tests can be integrated as part of the continuous integration process through deployment automation and test virtualization
- Regression test results are made available to the entire team to shorten resolution



# Focus more of your testing on exploratory testing

- Record your manual test procedures in English
- Turn on your recorder to log your exploratory testing
- Facilitate your defect submission

Manual Steps ?

Step	Description:	Expected Results	Validates
1	select the "Jazz Community Site" window		
2	select the "Rational Quality Manager - Projects - Jazz Community Site" window		
3	select the "New Work Item - Change and Configuration Management" window		
4	click the "Conditions" link		
5	drag the "com_ibm_team_workitem_web_ui_internal_view_queryeditor_Attribute" html		
6	click the "is" textbox		
7	click the "Run" button		

[Click to add step](#)

Keyword View ?

Clipboard ?

Type Filter Text

Steps

- Insert New Step Above
- Insert New Step Below
- Insert New Step(s) via Recording
- Reporting Step
- Add to Clipboard
- Remove Step
- Link to any Requirement
- Remove Requirement Links from step

# New offerings, real return on investment

## INTERNATIONAL PAPER

- After an acquisition, needed to get off rented infrastructure
- Move to webMethods as fast as possible
- Regression testing essential
- Stubbing of systems while they move over critical systems
- GH Tester performed all required functions quickly and easily
- Fully integrated in six months, two months early
- Saved significant rental costs

## Leading global financial services firm, assets of \$2 trillion+

- Bought next generation payments system
- Impact = organizational heart transplant
- Disparate, legacy formats
- Stubbed third party systems, otherwise unavailable for testing
- Reduced 10 days of manual testing to 10 minutes
- Saved >\$7 million so far
- “Project would have been impossible without the tool”

## €30 billion international supermarket operator

- Upgrade to webMethods 8
- “You need 30,000 hours to test the new environment”
- GH Tester + GCS = 4,000 hours
- Focus on business & volume critical components
- New testing strategy for all developments
- Cut time and costs but NOT quality

# The New IBM Rational Solution powered by Green Hat technology is designed to deliver measureable results and business value



## **Major telecom carrier**

Multiple test releases required additional test resources, increasing testing costs

- Solution: Virtualized services and created reusable test cases across multiple environments
- Results: Reduced testing time by 50%, saving \$800K over 3 years



## **A leading global financial services firm with assets of over \$2 trillion**

Many disparate legacy format to be migrated to next gen payments system

- Solution: Virtualized third-party systems otherwise unavailable for testing
- Results: Reduced manual testing from 10 days to 10 minutes, saving over \$7 million to date



## **Major U.S. insurer**

Recognized manual efforts insufficient for complex SOA and web services

- Solution: Agile middleware solution created to match the legacy systems' functionality
- Results: Reduced testing by 95% to 2 hours; reduced 'rate filing' validation by 94% to 320 hours



## **Global manufacturer acquires competitor**

Migrate off rented infrastructure onto company's standardized middleware platform

- Solution: Virtualized critical, unavailable systems during migration
- Results: Fully integrated in 6 months – 2 months ahead of schedule; saved significant rental costs and dependencies on third-party systems



# IBM Rational Test Virtualization Solution

Powered by Green Hat Technology

IBM Software

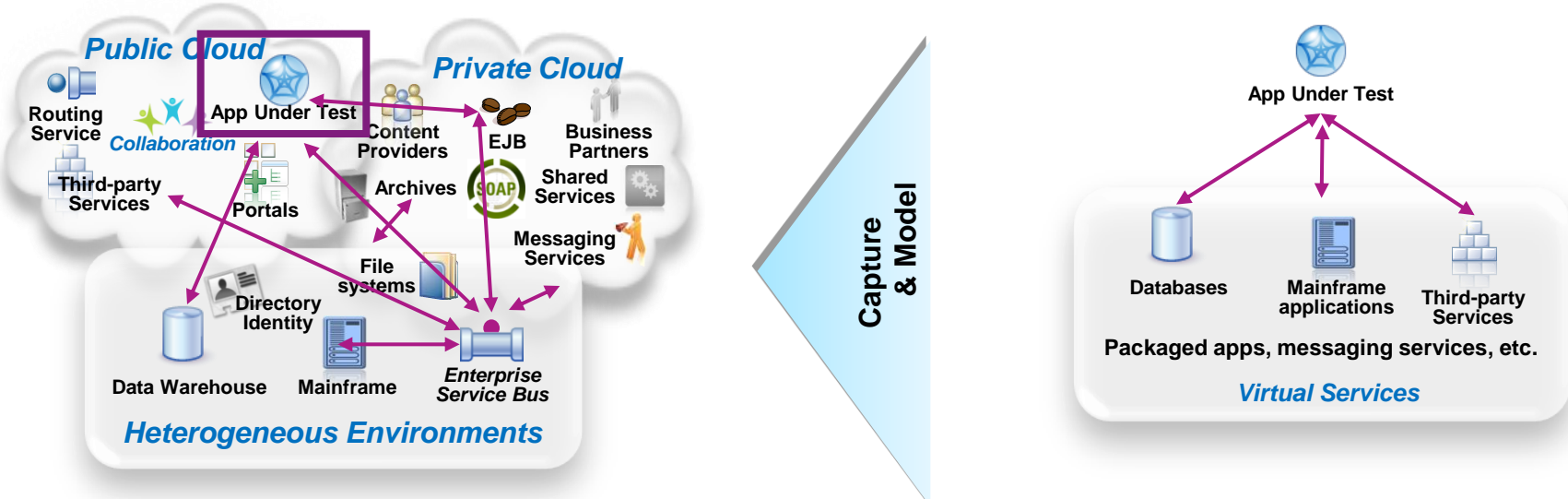
# Innovate2012

The Premier Event for Software and Systems Innovation





# What is Test Virtualization?



**System dependencies is a key challenge in setting up test environments:**

- ▶ **Unavailable/inaccessible:** Testing is constrained due to production schedules, security restrictions, contention between teams, or because they are still under development
- ▶ **Costly 3rd party access fees:** Developing or testing against Cloud-based or other shared services can result in costly usage fees
- ▶ **Impractical hardware-based virtualization:** Systems are either too difficult (mainframes) or remote (third-party services) to replicate via traditional hardware-based virtualization approaches

**Test Virtualization enables to create “virtual services”:**

- **Virtual Services simulate the behavior of an entire application or system during testing**
- **Virtual Services can run on commodity hardware, private cloud, public cloud**
- **Each developer, tester can easily have their own test environment**
- **Developer and testers continue to use their testing tools (Manual, Web performance, UI test automation)**

## Why IBM Rational Test Virtualization Solution ?

*A Smarter Solution for Better Quality*

### Significantly Lesser Test Lab costs

- Test lab infrastructure costs can be reduced by up to 90%
- Labor involved in setting up test environments can be reduced by 80%+
- Reduced or eliminate the cost of invoking 3rd party systems for non-production use, fee-based web services

### Reduced Cycle Time

- Test environments can be configured in minutes vs weeks
- More testers can be focused on testing, rather than configuring test environments
- More regression testing can be done independently from the User Interface, during development

### Lower Risk

- Developers have the means to test software earlier at the Service/API level
- Large teams working on different parts of an application or system can effectively do parallel development by virtualizing different parts of the system

# The New IBM Rational Solution

## *Powered by Green Hat Technology*



### ■ IBM Rational Test Workbench

- Enable functional, performance and integration testing throughout your project lifecycle.
- With a script less, wizard-driven test authoring environment and support for more than 70 technologies and protocols,

### ■ IBM Rational Performance Test Server

- Save time and effort by reusing your functional integration tests for performance testing
- Combine your tests to model real-world scenarios and assess the impact of load on your integrated infrastructure.

### ■ IBM Rational Test Virtualization Server

- Model real system behavior to eliminate test dependencies and simulate production,
- Accelerate testing and reduce the setup and infrastructure costs of traditional testing environments.



QA staff

Test Workbench	Functional Testing Integration Testing Performance Testing Regression Testing
----------------	--

Test Servers	Load Agents Virtualization Agents
--------------	--------------------------------------

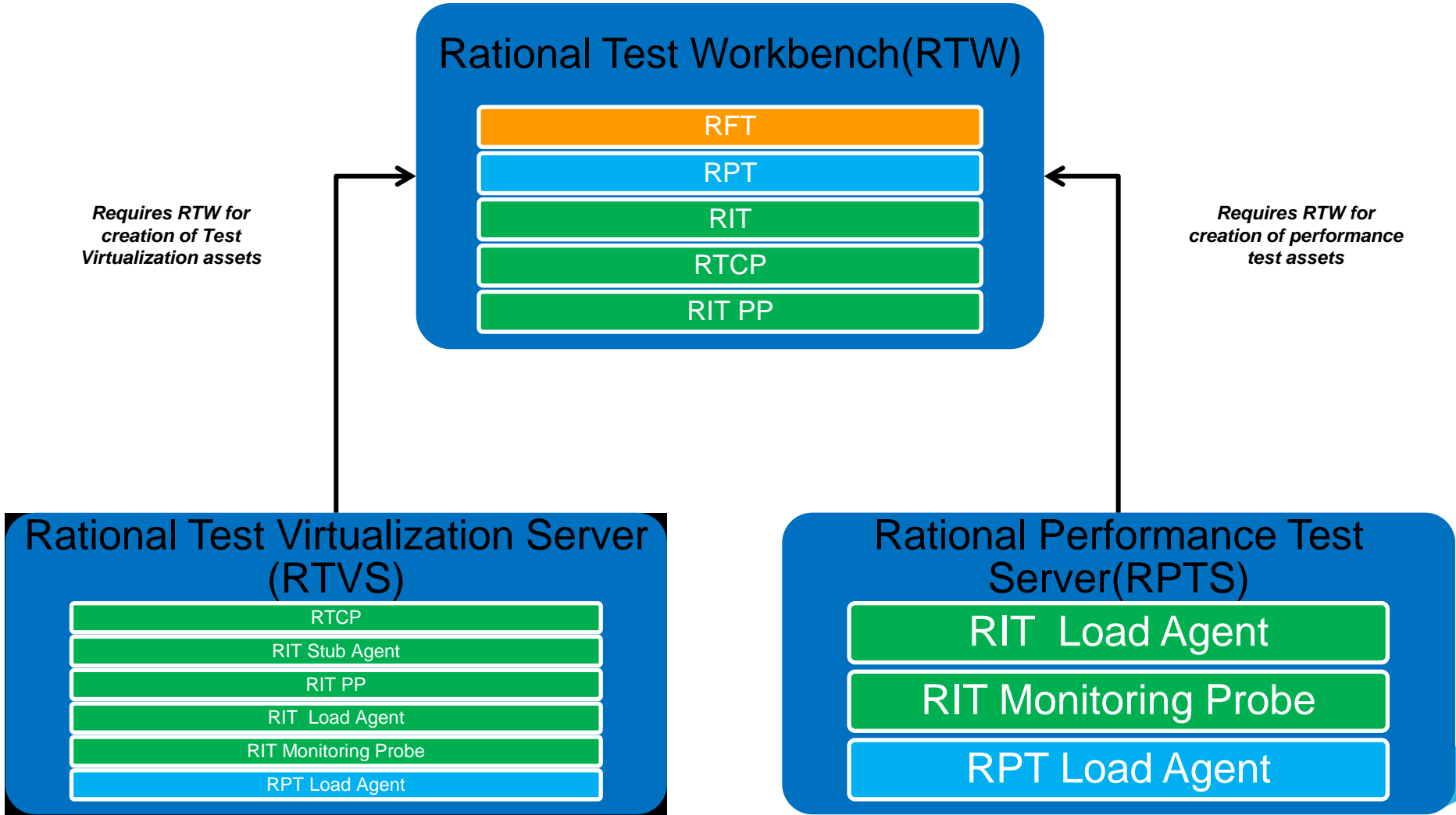


Test Lab Infrastructure

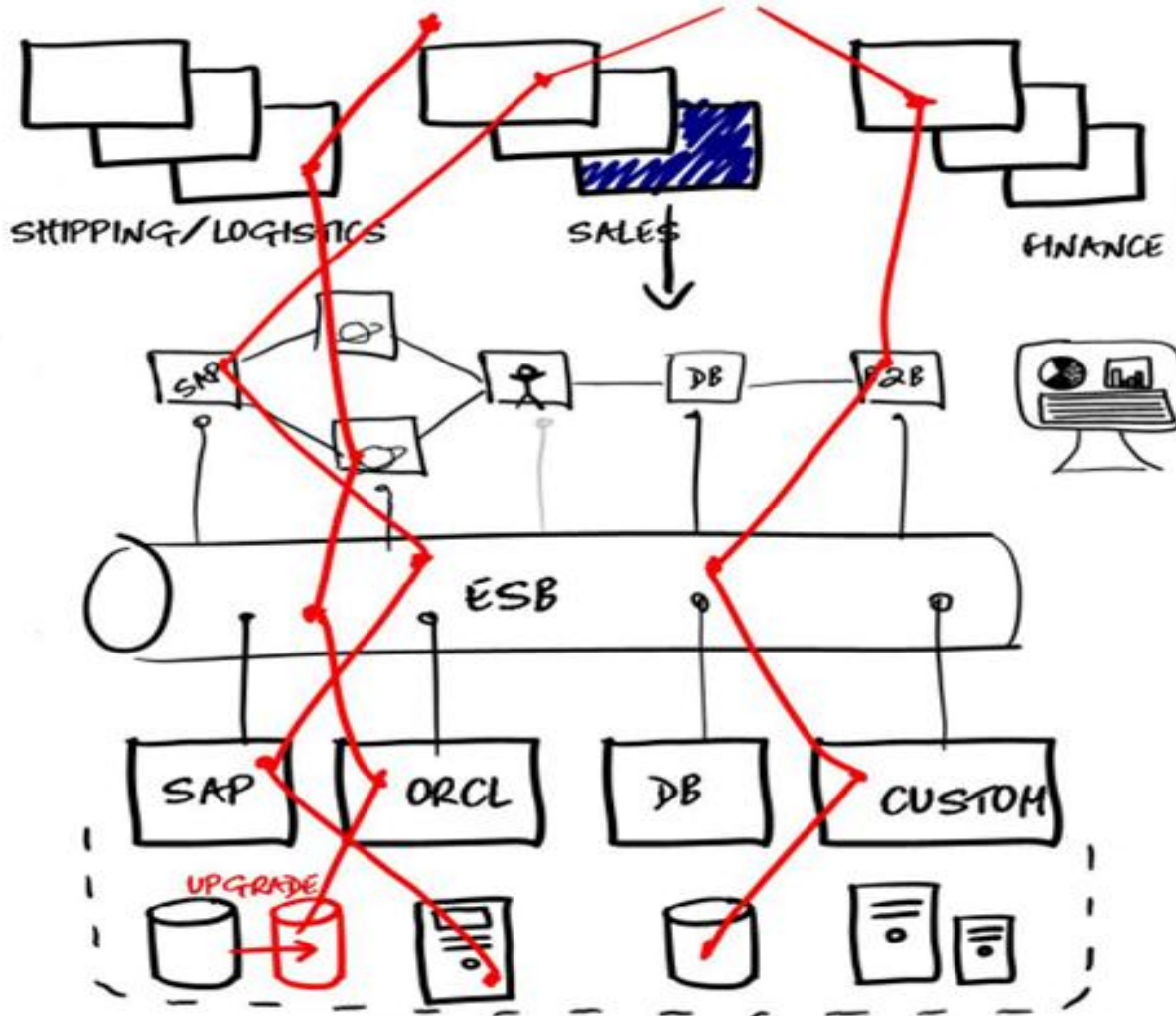
*Launched June 2012*



# The Solution Ingredients



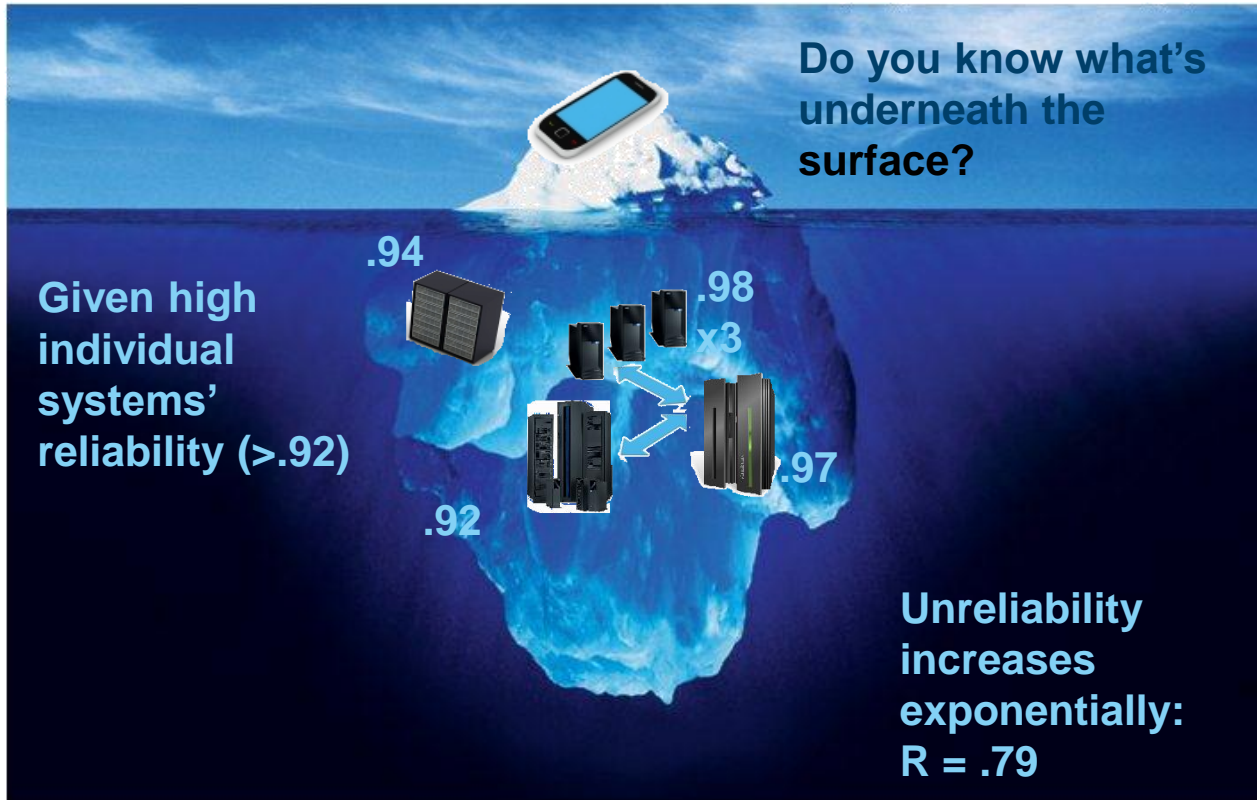
# Complexity.... Made Simple!



- Multiple layers of technology
- Multiple vendor platforms
- Complex transactions
- Complex dependencies
- Multiple stakeholders

## Taking a new look at integration testing

- Integrating highly reliable systems doesn't guarantee high reliability overall
- Testing below the surface is critical in increasingly complex systems

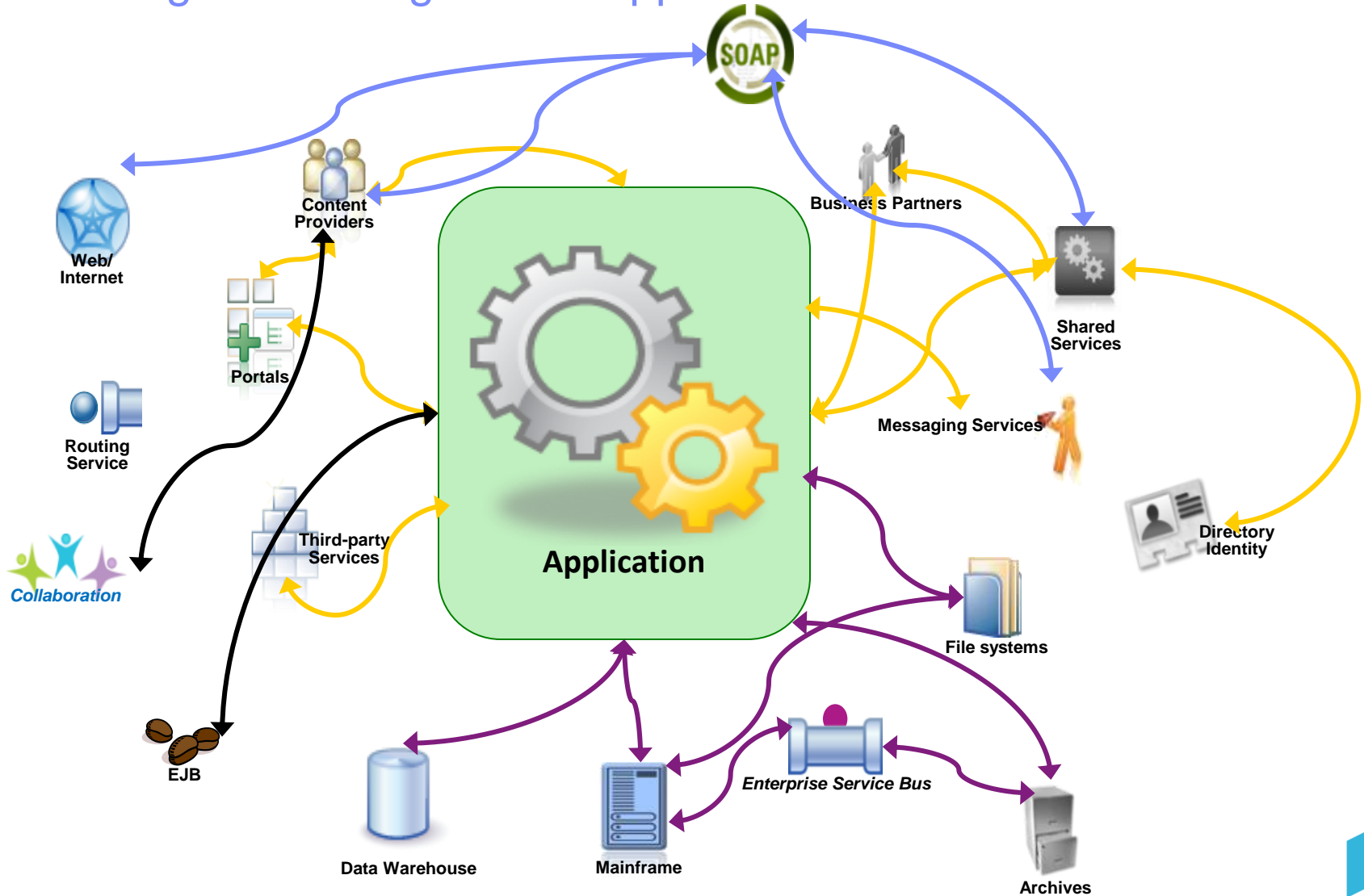


***“In practice, integration has proven to be the most enlightening activity in exposing the risks and uncertainties in software development.”\*\****

\*\*Walker Royce, Chief Software Economist at IBM



# Hard to configure heterogeneous apps



## Blockers for End to End testing

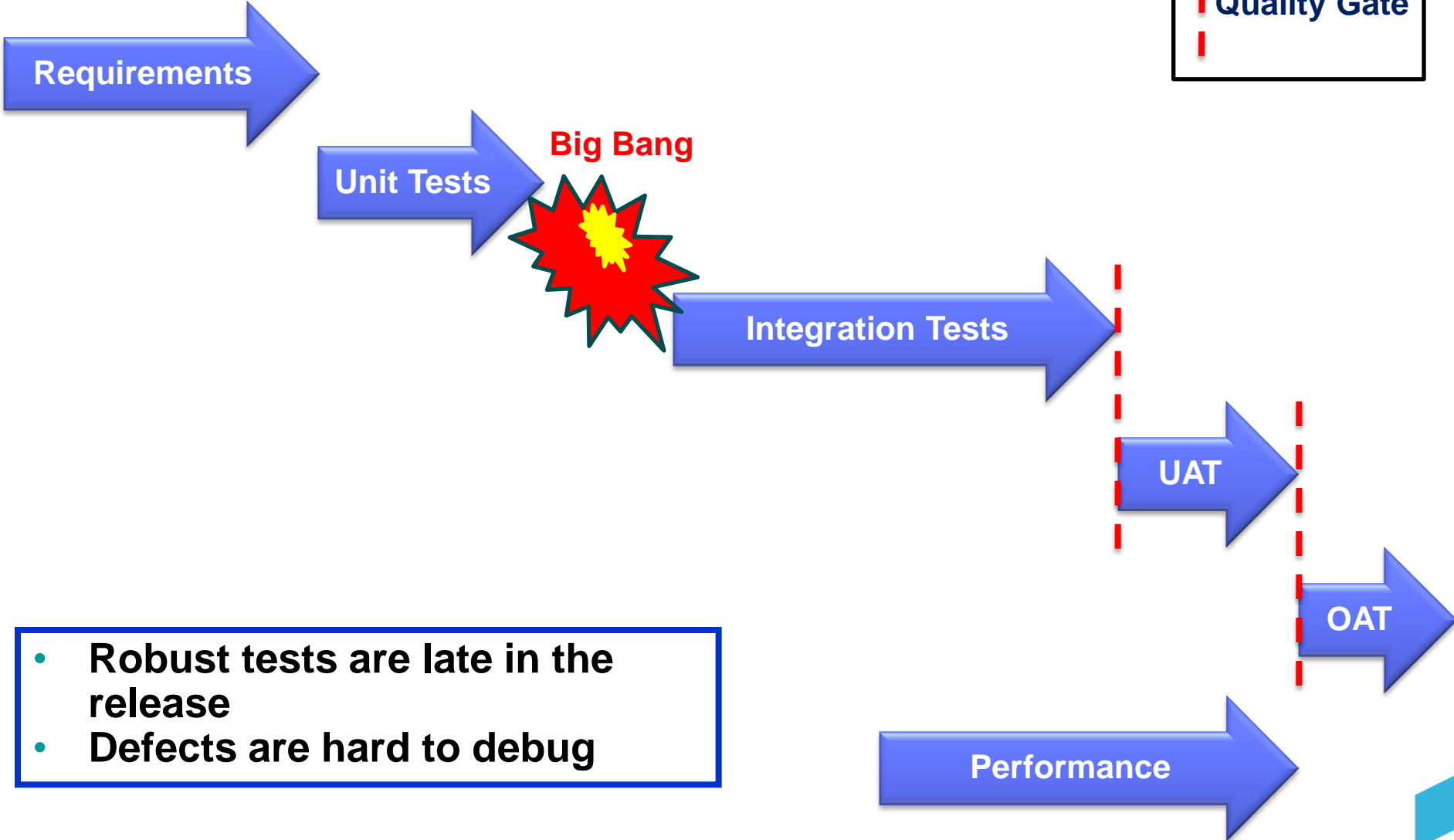
**"Too costly to setup a test message feed for test"**

**"We're still waiting for them to deploy"**

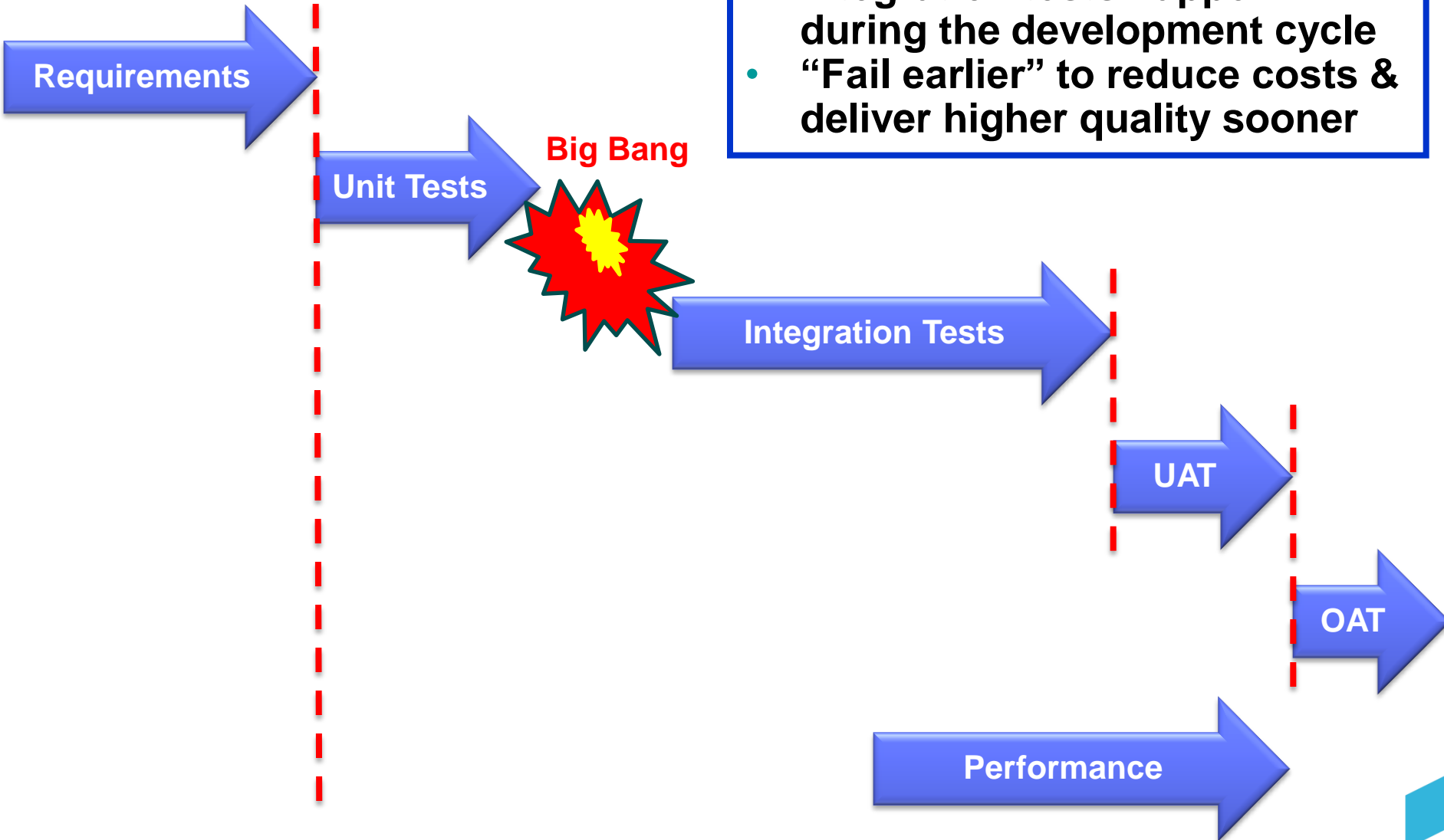
**"The environment will be another 3 weeks"**

**"The dev team hasn't started work on it yet"**

# The Old World

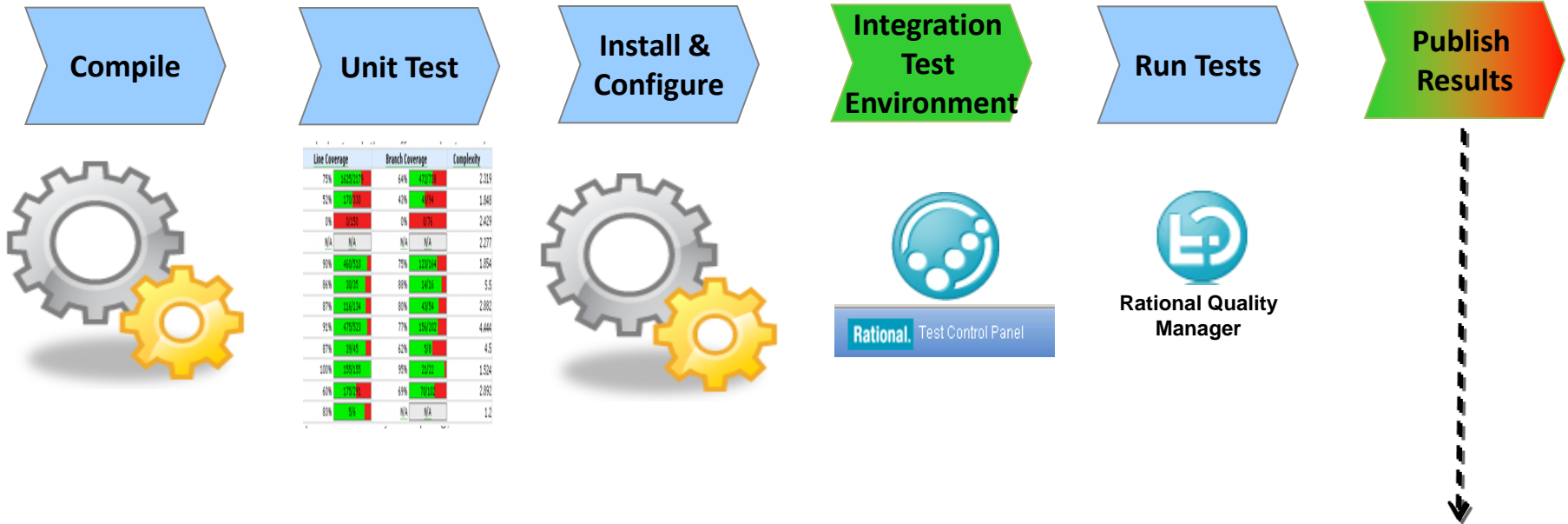


# The New World



# Continuous Integration testing for builds

**“Green” builds have passed meaningful integration tests**



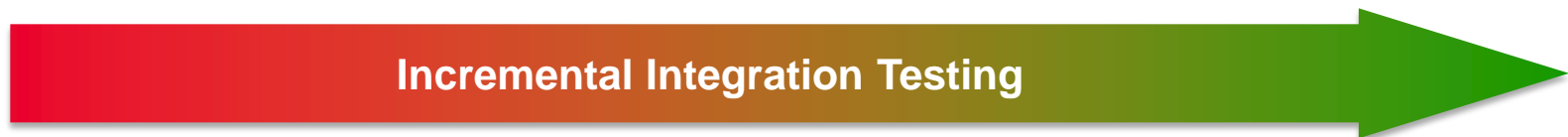
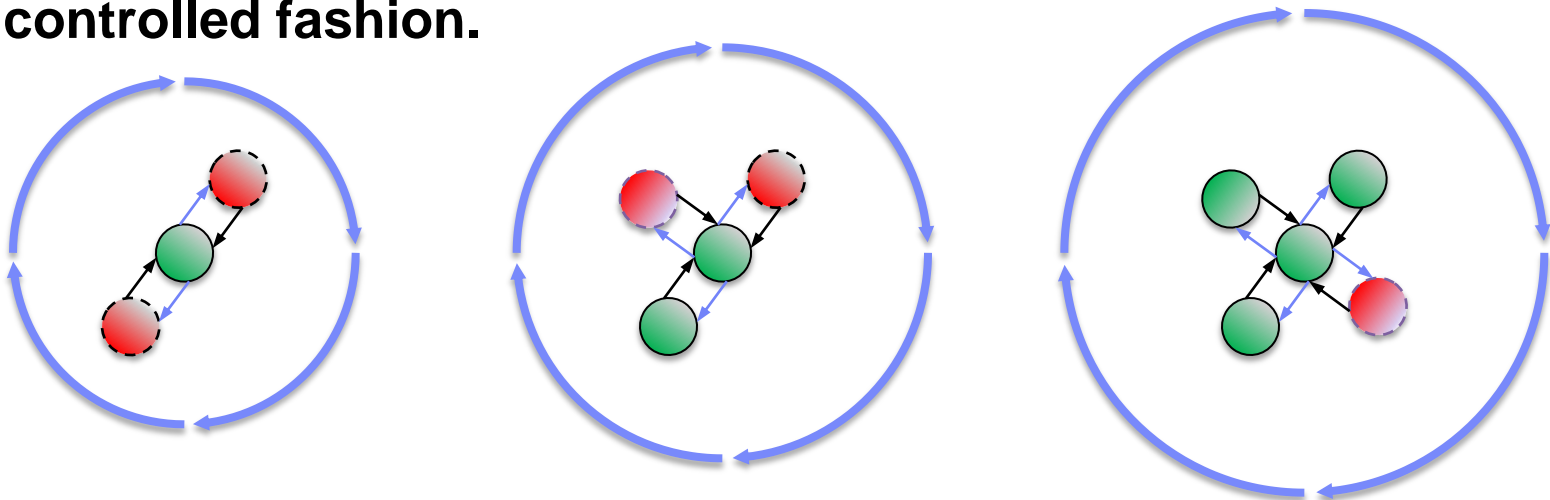
**Nightly build is not published until integration tests run.**

## System Under Test is evolving over time

✓ **Test Virtualization enables Continuous Integration Testing**

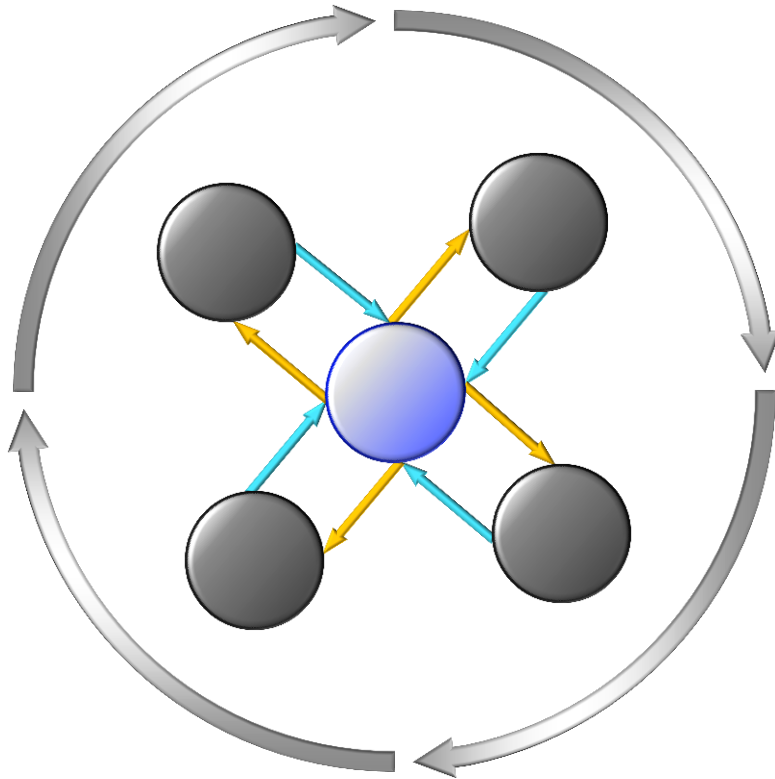
✓ **Services, applications, systems are introduced into the continuous integration cycle in a prioritized, controlled fashion.**

● **Actual Service/App**  
● **Virtual Service/App**







# Incremental Testing

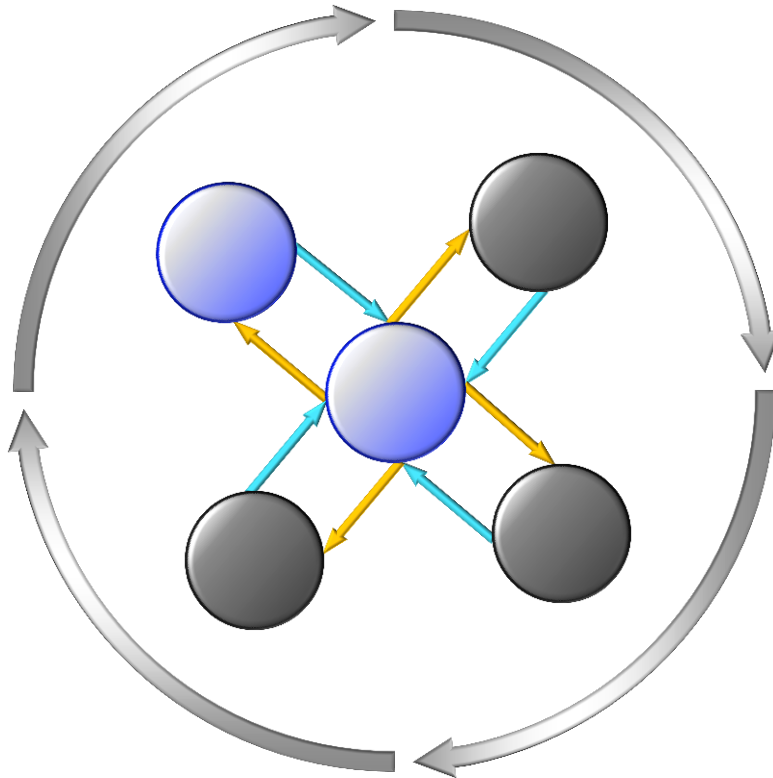




## Initial Stage:

**A single component can be tested in the context of an end to end environment, supplied by a set of virtualized components.**

-  **Actual Component**
-  **Virtualized Component**

## Incremental Testing

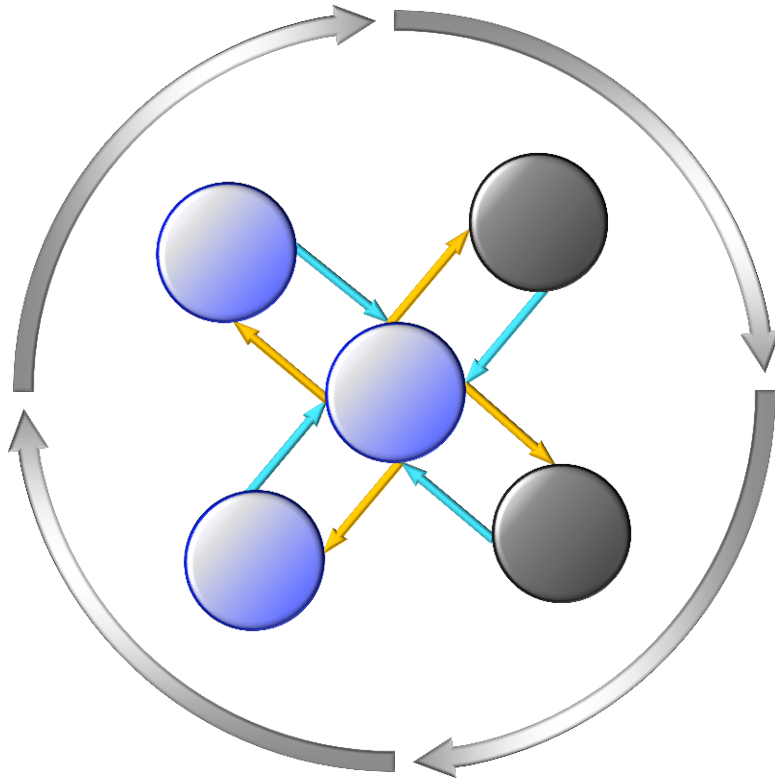




-  **Actual Component**
-  **Virtualized Component**

**As components are built:**

**The same end to end tests can be run, replacing virtualized components with actual components.**

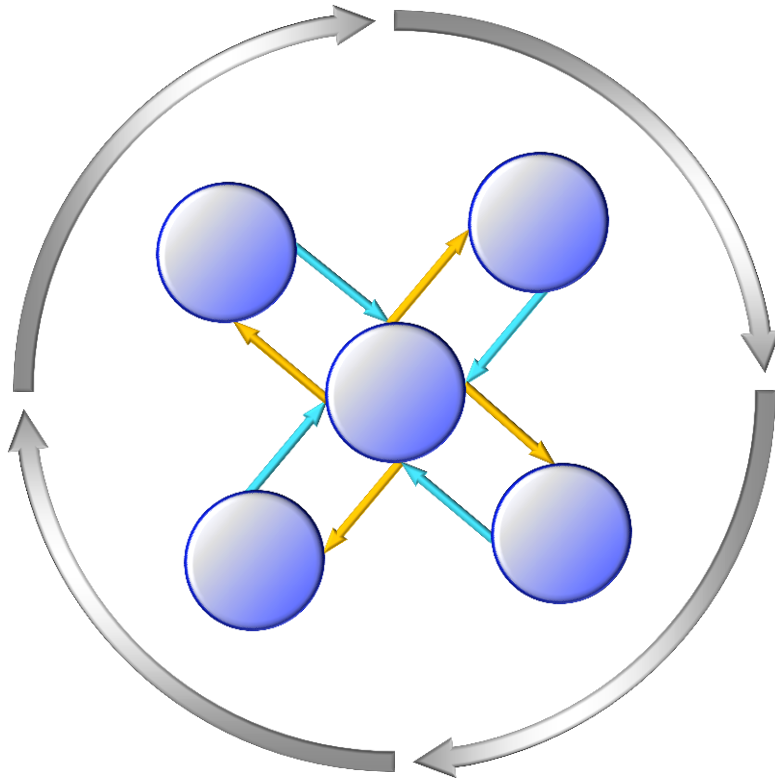
## Incremental Testing





-  **Actual Component**
-  **Virtualized Component**

**As components are built:**  
**This enables us to test downstream dependencies as they are built.**

## Incremental Testing

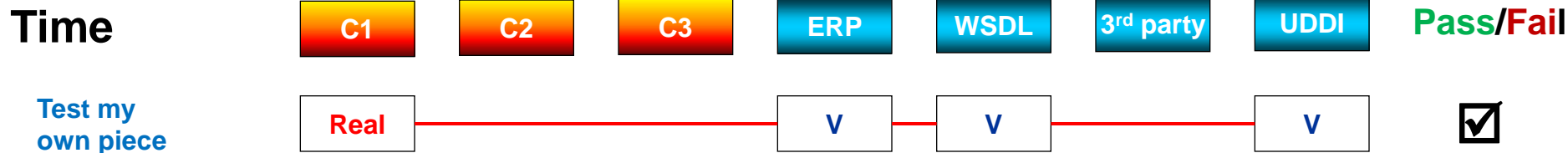


-  **Actual Component**
-  **Virtualized Component**

**When the system is complete:**  
**End to end testing can be carried out  
with fewer surprises and lower risk**

# Continuous Integration Testing with Test Virtualization

- Integration Testing requires components that may not be ready/available yet, or expensive to use - Test Virtualization enables replacing them with a virtual component.
- Services, applications, systems are introduced into the continuous integration cycle in a prioritized, controlled fashion.



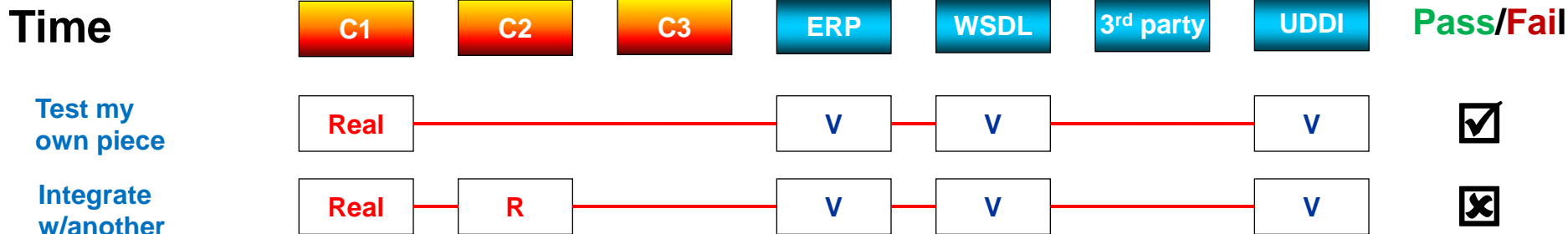
**Example:**

- Test C1 with three virtualized services.
- Can use simple or complex integration scenarios.
- Quick to setup and low-cost.

*By SUT I also mean custom code...*

# Continuous Integration Testing with Test Virtualization

- Integration Testing requires components that may not be ready/available yet, or expensive to use - Test Virtualization enables replacing them with a virtual component.
- Services, applications, systems are introduced into the continuous integration cycle in a prioritized, controlled fashion.

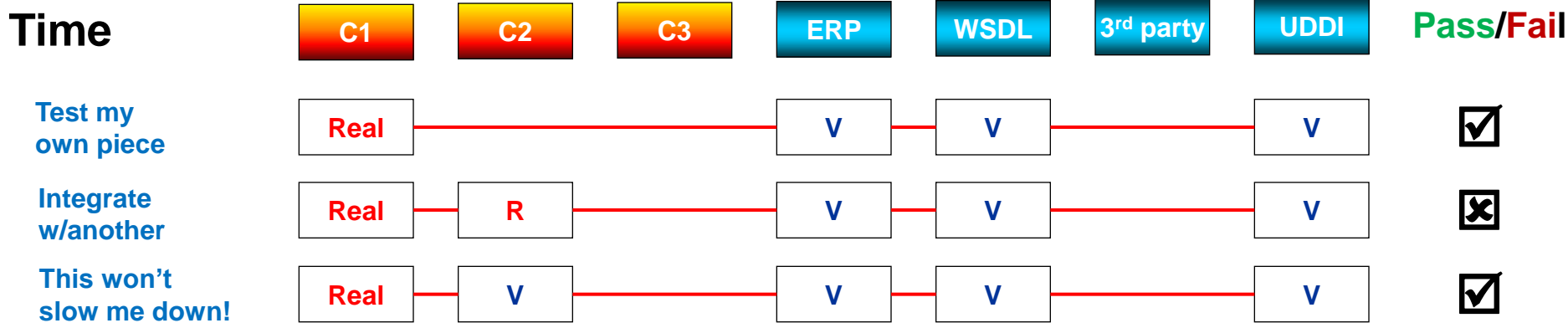


- Integrate with C2...
- Can use more complex integration scenarios.
- Quick to setup and low-cost.
- OOPS !



# Continuous Integration Testing with Test Virtualization

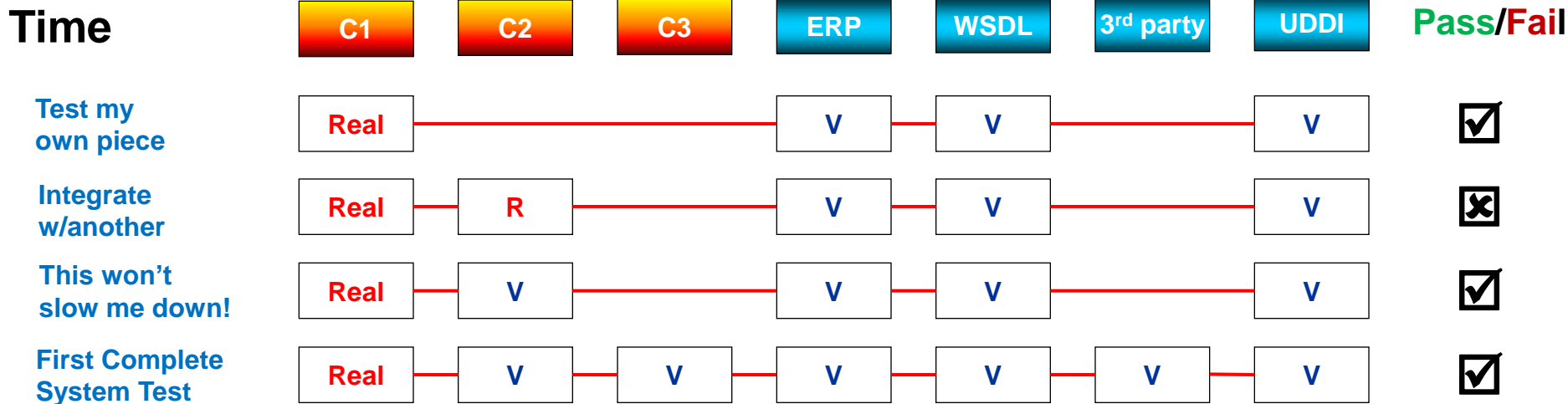
- Integration Testing requires components that may not be ready/available yet, or expensive to use - Test Virtualization enables replacing them with a virtual component.
- Services, applications, systems are introduced into the continuous integration cycle in a prioritized, controlled fashion.



- C2 introduced some defects – replace it with a virtual service!
- A defect in C2 doesn't stop testing of those who depend on it!
- Quick to setup and low-cost.

# Continuous Integration Testing with Test Virtualization

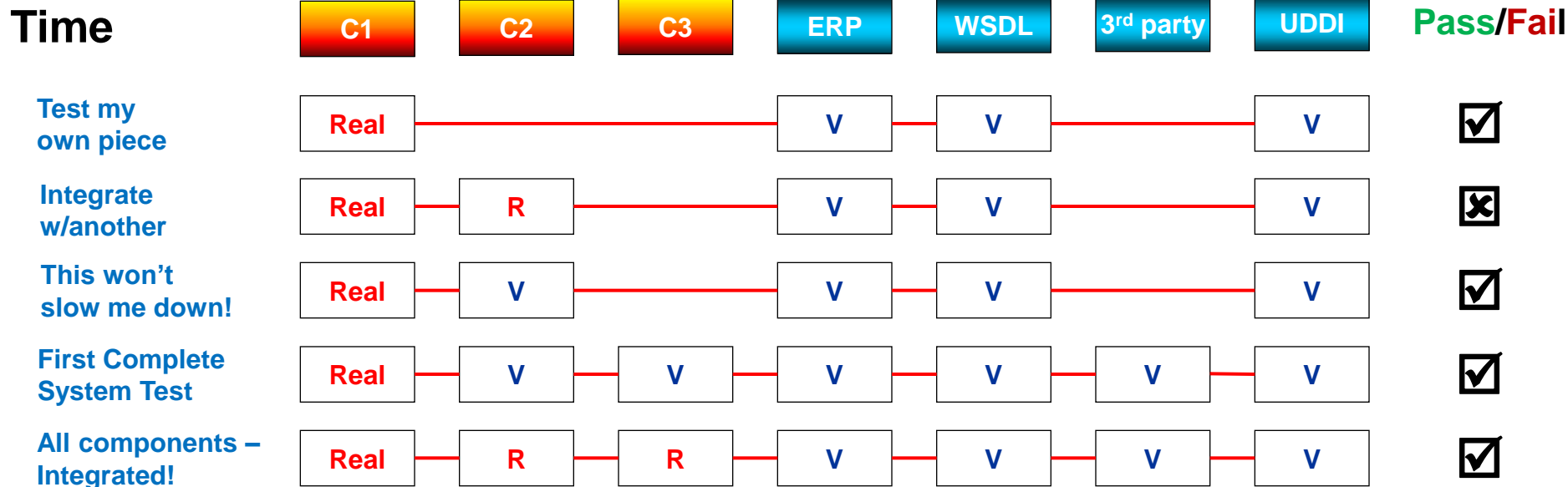
- Integration Testing requires components that may not be ready/available yet, or expensive to use - Test Virtualization enables replacing them with a virtual component.
- Services, applications, systems are introduced into the continuous integration cycle in a prioritized, controlled fashion.



- First complete system test: Everything but C1 is Virtual!
- Enables Testing with enterprise integration scenarios.
- Add testing scenarios to Build Verification Testing (BVT)
- Quick to setup and low-cost.

# Continuous Integration Testing with Test Virtualization

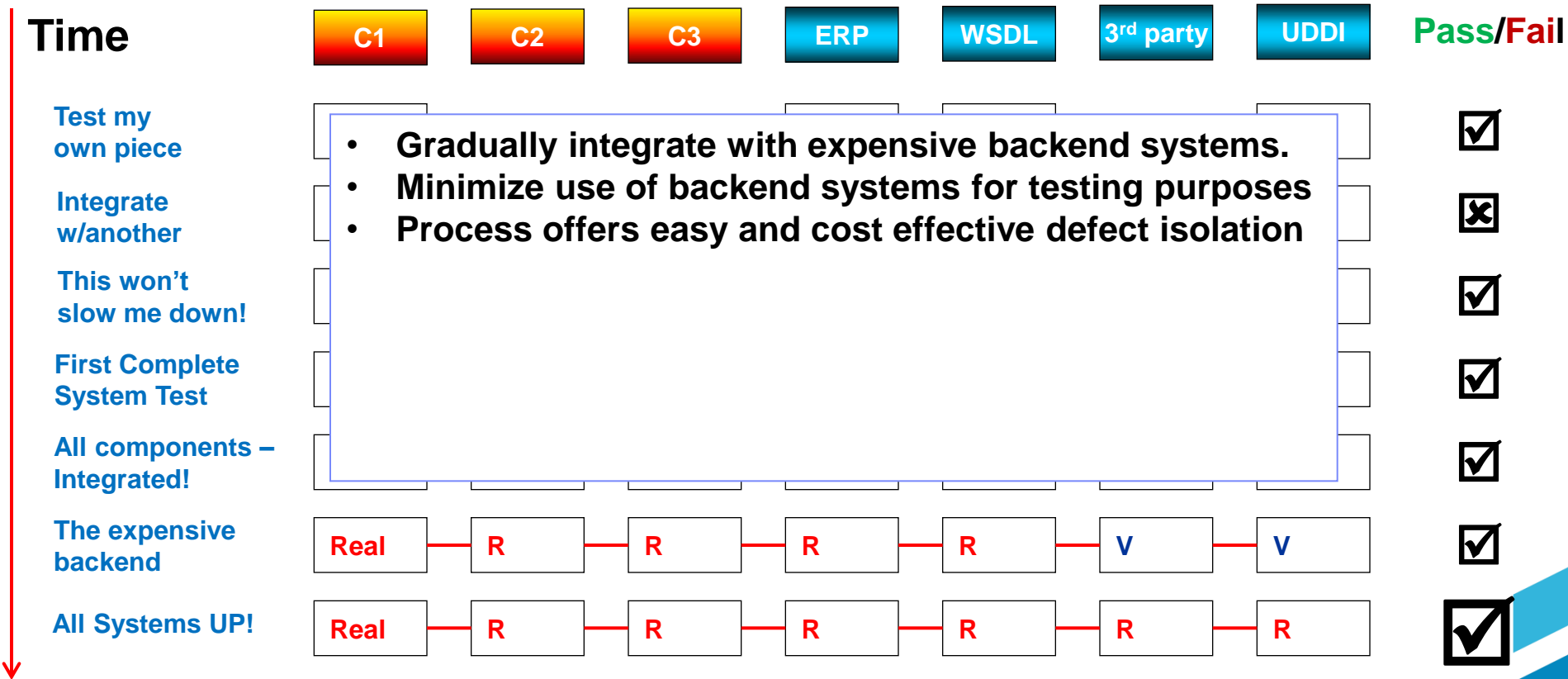
- Integration Testing requires components that may not be ready/available yet, or expensive to use - Test Virtualization enables replacing them with a virtual component.
- Services, applications, systems are introduced into the continuous integration cycle in a prioritized, controlled fashion.



- Integrate C2 and C3... with virtualized backend systems
- Use enterprise integration scenarios for Testing, inc BVT!
- Quick to setup and low-cost.

# Continuous Integration Testing with Test Virtualization

- Integration Testing requires components that may not be ready/available yet, or expensive to use - Test Virtualization enables replacing them with a virtual component.
- Services, applications, systems are introduced into the continuous integration cycle in a prioritized, controlled fashion.



## Supported Environments & Technologies

### Messaging Protocols

- ActiveMQ
- Email (SMTP, IMAP)
- Files
- FTP/S
- HTTP/S
- JMS (JBOSS et al)
- IBM WebSphere MQ
- JBoss MQ
- SAP IDoc, BAPI, RFC & XI/PI
- Software AG's IB & IS
- Solace
- Sonic MQ
- TCP
- TIBCO Rendezvous, Smart Sockets & EMS
- Custom

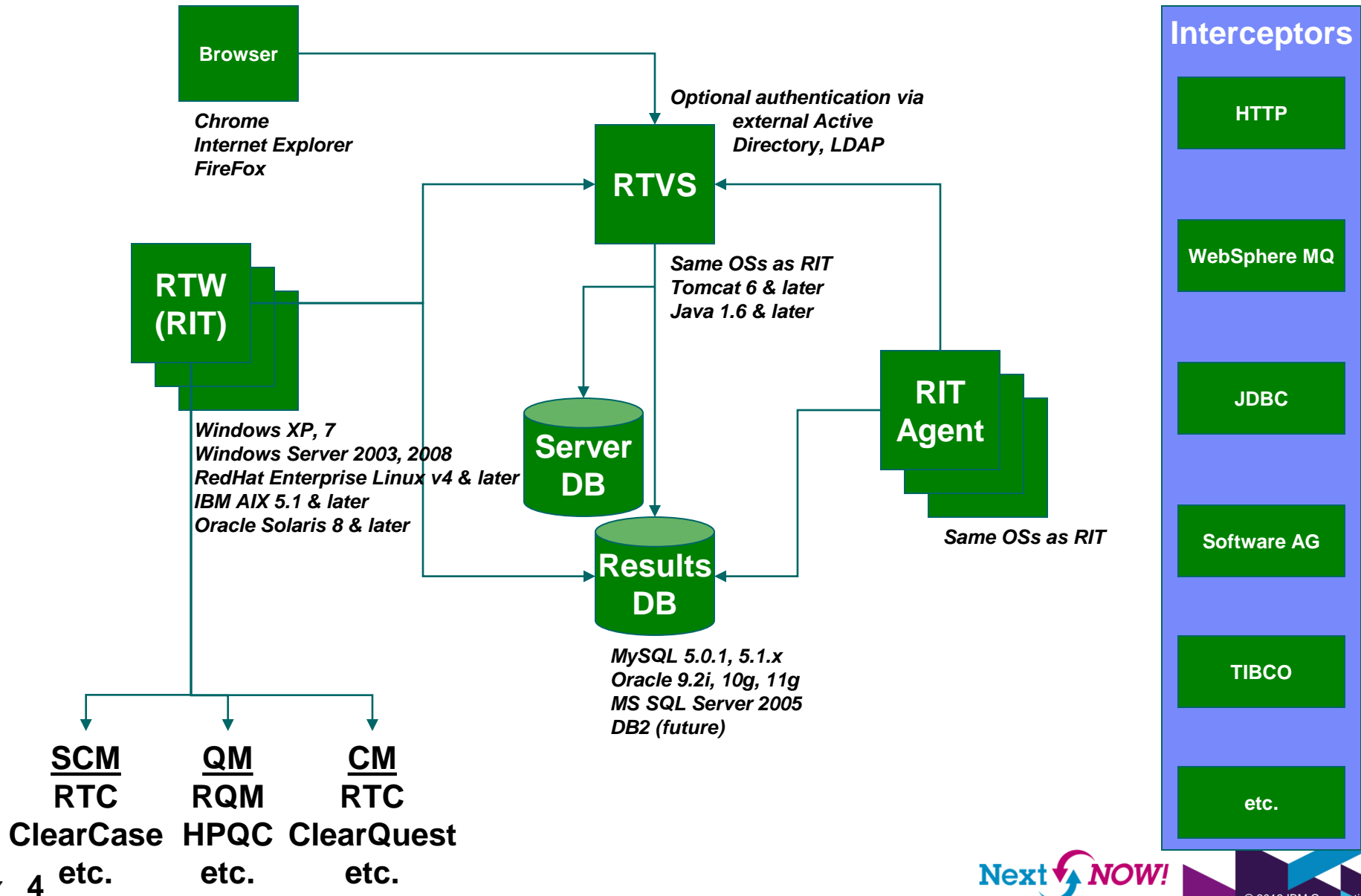
### SOA, ESB, Others

- CentraSite
- Oracle Fusion
- SCA Domain
- Software AG IS, BPMS
- Sonic ESB
- TIBCO ActiveMatrix
- UDDI
- Web Services
- WebSphere RR
- WSDL
  
- BPM
- Databases
- Log Files

### Message Formats

- .Net Objects
- Bytes
- COBOL Copybook
- ebXML
- EDI
- Fixed Width
- HL7
- IATA
- Java Objects
- MIME
- OAG
- SOAP
- Software AG Broker Docs
- SWIFT
- TIBCO ActiveEnterprise
- XML (DTD, XSD, WSDL)
- Custom

# Deployment Architecture







## Summary

*A Smarter Solution for Better Quality*



IBM Software

# Innovate2012

The Premier Event for Software and Systems Innovation



# The New IBM Rational Solution

## *Powered by Green Hat Technology*



### ■ IBM Rational Test Workbench

- Enable functional, performance and integration testing throughout your project lifecycle.
- With a script less, wizard-driven test authoring environment and support for more than 70 technologies and protocols,

### ■ IBM Rational Performance Test Server

- Save time and effort by reusing your functional integration tests for performance testing
- Combine your tests to model real-world scenarios and assess the impact of load on your integrated infrastructure.

### ■ IBM Rational Test Virtualization Server

- Model real system behavior to eliminate test dependencies and simulate production,
- Accelerate testing and reduce the setup and infrastructure costs of traditional testing environments.



QA staff

Test Workbench	Functional Testing Integration Testing Performance Testing Regression Testing
----------------	--

Test Servers	Load Agents Virtualization Agents
--------------	--------------------------------------



Test Lab Infrastructure

*Launched June 2012*



# IBM Rational Test Virtualization Solution

*A Smarter Solution for Better Quality*

## Significantly Lesser Test Lab costs

- Test lab infrastructure costs can be reduced by up to 90%
- Labor involved in setting up test environments can be reduced by 80%+
- Reduced or eliminate the cost of invoking 3rd party systems for non-production use, fee-based web services

## Reduced Cycle Time

- Test environments can be configured in minutes vs weeks
- More testers can be focused on testing, rather than configuring test environments
- More regression testing can be done independently from the User Interface, during development

## Lower Risk

- Developers have the means to test software earlier at the Service/API level
- Large teams working on different parts of an application or system can effectively do parallel development by virtualizing different parts of the system



[www.ibm.com/software/rational](http://www.ibm.com/software/rational)

© Copyright IBM Corporation 2012. All rights reserved. The information contained in these materials is provided for informational purposes only, and is provided AS IS without warranty of any kind, express or implied. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, these materials. Nothing contained in these materials is intended to, nor shall have the effect of, creating any warranties or representations from IBM or its suppliers or licensors, or altering the terms and conditions of the applicable license agreement governing the use of IBM software. References in these materials to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates. Product release dates and/or capabilities referenced in these materials may change at any time at IBM's sole discretion based on market opportunities or other factors, and are not intended to be a commitment to future product or feature availability in any way. IBM, the IBM logo, Rational, the Rational logo, Telelogic, the Telelogic logo, and other IBM products and services are trademarks of the International Business Machines Corporation, in the United States, other countries or both. Other company, product, or service names may be trademarks or service marks of others.